International Journal of Innovative
Computing, Information and Control
Volume 11, Number 5, October 2015

 ISSN 1349-4198
pp. 1509–1523

MIXED GAUSSIAN AND IMPULSE NOISE REMOVAL BASED ON KERNEL OBSERVATION AND EDGE DIRECTION

Fitri Utaminingrum[1,*], Keiichi Uchimura[2] and Gou Koutaki[3]

[1]Faculty of Computer Science
Brawijaya University
Jl. Veteran No. 8 Malang, East Java, Indonesia
*Corresponding author: f3_ningrum@yahoo.com

[2]Graduate School of Science and Technology
[3]Priority Organization for Innovation and Excellence
Kumamoto University
2-39-1 Kurokami, Chou-ku, Kumamoto 860-8555, Japan

Received November 2014; revised April 2015

Abstract. *Two stages filtering process to reduce the mixed Gaussian plus impulse noise is proposed. The first stage is reducing impulse noise. Furthermore, the second stage is reducing Gaussian noise. A merger between the decision based method and kernel observation is conducted to reduce impulse noise. Meanwhile, five Gaussian masks are used to reduce Gaussian noise. The use of the mask is adjusted to the image texture. In contrast to the previous method, our method can give a much better restoration in some particular cases, not only in the quality of the filtering result, but also in the fast computation time. In this study, we can improve the speed of previous method. Experimental results have shown that the proposed method outperforms the alternative filters in suppressing the mixed Gaussian plus impulse noise.*
Keywords: Gaussian noise, Impulse noise, Filter, Kernel observation

1. **Introduction.** The corrupted image with noise is one of the main problems in the image processing and computer vision [1]. Noise may appear during any of acquisition, pre-processing, compression, production phase, storage and transmission. In transmission signal, the signal received at the receiver side weakened. For example, the distance between transmitter and receiver of the television is located far apart, so that the received signal at the receiver side of the television becomes weak and noise will appear. Two common types of noise which usually appear are Gaussian noise and impulse noise. Currently, the development of digital television technology is rapid that is following many people's needs. The digital television is able to implement a filter algorithm that has optimal performance in eliminating noise. Mostly, the existing image denoising algorithm is only optimal to reduce one type of noise (Gaussian or impulse noise). However, the digital image is often suffered from more than one type of noise during the process of acquisition, pre-processing, compression and transmission [2, 3].

Denoising method that is only used for reducing one type of noise is much easier than the mixed noise removal. In such adverse conditions, it needs the method to press noise in the image to become a better-quality image. Contamination image with noise makes a user difficult to recognize the original image data.

The best performance of the filtering technique is accomplished by performing iteratively the noise filtering process until the several pixels detected as noisy become negligibly small [4]. A good image denoising model has significant characteristics of removing noise

while preserving edges details of an image [5]. In addition, it should have low computational complexity and fast computation time process [6].

A large number of algorithms have been proposed to remove mixed Gaussian plus impulse noise, some of which are based on Total Variation (TV) regularization [7, 8, 9, 10, 11]. TV regularized energy minimization [12] performs well for preserving edges area while removing noise and has been used for multiplicative noise removal [13, 14, 15]. However, TV regularized often causes staircase effect [16] and the texture information over-smooth in the image [17].

Sparse land model and K-Singular Value Decomposition (K-SVD) algorithms [18] are also used to reduce Gaussian noise [19]. Furthermore, modifying K-SVD is used to optimize the mixed Gaussian and impulse noise removal. Modifying K-SVD is combining the impulse noise removal based on the mean of the neighboring uncorrupted pixels and an effective learning dictionary method. Pressing of an impulse noise in the early stages is able to optimize the performance of the K-SVD [20]. The filtering result has a good visual image. However, computation time is not so fast, and KSVD method can be used for filtering a single image that is obtained from photo satellite and medical imaging application. Therefore, the applications of K-SVD in image sequences are still needed to be improved, especially in computation time problem. Application of K-SVD method must be supported by high-speed processors. Therefore, it is necessary to design a new image filtering method that is easy to be implemented, and it has a fast computing time process. However, it has capabilities almost similar to the K-SVD method.

There are many goals in designing noise removal methods.

1. The visual result of the proposed method must have a smooth, clear texture and no artefacts in the filtering result.
2. The important texture detail should not be lost.
3. The filtering process is conducted especially on the noisy pixel, without engaging the important pixel that is indicated as original pixel.
4. Image boundaries should not be blurred or sharpened.
5. Preserve the integrity of edge area.
6. An approach of the filtering method is created to obtain a fast computation time.

In this paper, we propose a new filtering method based on kernel observation and edge direction. This method is the development of the previous method which only used a mask in the filtering process and did not pay attention to the edge of the image area. In our proposed method, we use kernel observation method for reducing impulse noise and adaptive kernel selection based on an edge image to suppress Gaussian noise. This paper provides five types of Gaussian mask based on the direction of the edge of the image. The writings of this paper are organized from Section 1 up to Section 5.

2. Noise in Image.

The digital image is often disturbed by noise. In this paper, a filter is conducted to reduce the mixed Gaussian and impulse noise. We provide a corrupted image as the input of the filter that we made. We create the corrupted image by mixed Gaussian and impulse noise by performing the following steps.

- Firstly, we corrupt the clean image ($\mathbf{F}$) with Gaussian noise ($\mathbf{g}$). Hence, the resulting image is the corrupted image by Gaussian noise ($\mathbf{I}$), as described in Equation (1).

$$\mathbf{I} = \mathbf{F} + \mathbf{g} \tag{1}$$

- Secondly, furthermore, we mix a corrupted image by Gaussian noise ($\mathbf{I}$) with impulse noise as illustrated in Equation (2).

$$\hat{I}_{ij} = \begin{cases} d_{ij} & \text{with probability } p \\ I_{ij} & \text{with probability } 1-p \end{cases} \tag{2}$$

$\hat{I}_{ij}$ is the element of corrupted image by mixed Gaussian and impulse noise. d_{ij} is uniformly distributed random numbers between the minimum $d_{\min}$ and the maximum value $d_{\max}$. p is the noise density of random-valued impulse noise with $0 \leq p \leq 1$.

3. **Proposed Method.** The mixed Gaussian and impulse noise is removed in our proposed method through two stages filtering process. The first stage is reducing impulse noise. Furthermore, the second stage is reducing Gaussian noise. The detail information about both stages of the filtering process is explained in the following subsection.

3.1. **Impulse noise removal.** We make impulse noise removal of a merger between the decision based method and the kernel observation. Decision based method is conducted for filtering pixels that are detected as impulse noise. Thus, the method needs impulse noise detectors. Meanwhile, the kernel observation inspects the uncorrupted pixels surrounding the pixel that was detected as impulse noise.

3.1.1. *Impulse noise detector.* In the proposed method, we introduce a simple impulse noise detector that can be used to detect the fixed impulse noise and random-value impulse noise. Detection of the fixed impulse noise such as salt and pepper noise is easier than random-value impulse noise. This is caused by the noise intensity which has fixed value at 0 or 255. Hence, a noisy pixel in this condition is indicated by value 0 or 255. While for detecting random-valued impulse noise, we use sample pixel window ($\mathbf{W}$). Firstly, we calculate the absolute value of the difference between each element window and the median value, as presented in Equation (3). Sometimes in the high-density impulse noise, the result of median value in the window is 0 or 255. If that happens, we replace the median value with 128.

$$\hat{W}_{ij} = |\text{median}(\mathbf{W}) - W_{ij}| \tag{3}$$

We compare the value of each element of the window $\mathbf{W}$ with a threshold value (τ), for determining each element of the window $\mathbf{W}$ noisy pixel or not. If an element in the window has an absolute value greater than the threshold value, then the pixel at (i, j) position in the window will be detected as noise (N) and vice versa. The threshold value (τ) is calculated from the average value between the minimum and the maximum absolute value of element of the $\hat{\mathbf{W}}$ window.

$$\tau = \frac{1}{2}\left(\min(\hat{\mathbf{W}}) + \max(\hat{\mathbf{W}})\right) \tag{4}$$

3.1.2. *Impulse noise removal based on decision.* Decision based method only filters on the pixels that are detected as noise. Referring to noise detector (N), we can write this method as in Equation (5).

$$\hat{P}_{ij} = \begin{cases} O_{ij} & \text{if } \hat{I}_{ij} > \tau \\ \hat{I}_{ij} & \text{others} \end{cases} \tag{5}$$

$\hat{P}_{ij}$ is the element of impulse noise removal. O_{ij} is the replacement pixel that was detected as impulse noise. O_{ij} is obtained by observing the surrounding pixels that are detected as impulse noise. We are only averaging two pixels, which are not detected as impulse noise to obtain one pixel O_{ij}.

Observation begins from horizontal, vertical, left-diagonal and right-diagonal. For example, the observations of two pixels in the horizontal direction at the position (i, j) in the image $\hat{\mathbf{I}}$ are $\hat{I}_{i,j-1}$ and $\hat{I}_{i,j+1}$. Further, by turning it on the angles $\angle 90°$, $\angle 45°$,

$\angle 135°$ pixels observations on the vertical direction, left-diagonal and right-diagonal will be obtained. Preliminary observations for two noises-free pixels are conducted in the 3×3 window. When observation could not find two noises-free pixel on the window 3×3 in the predetermined direction, the window size will be increased into 3×4, 4×4 and 4×5 successively.

In this stage, the image of $\hat{P}_{ij}$ does not contain impulse noise, but still contains the Gaussian noise. For that reason, the next stage needs the filtering process to reduce Gaussian noise.

3.2. **Gaussian noise removal.** Reducing Gaussian noise in the image can be conducted by convolution of the corrupted image with a Gaussian masks as in Equation (6).

$$\hat{\mathbf{F}} = \hat{\mathbf{P}} * \hat{\mathbf{H}}^n \tag{6}$$

$*$ is convolution operator. $\hat{\mathbf{P}}$ is the corrupted image with Gaussian noise and $\hat{\mathbf{H}}$ is Gaussian mask. n is type of Gaussian mask. In this paper, we provide five Gaussian masks. $\mathbf{H}^1$ $(n = 1)$ is symmetrical Gaussian mask. $\mathbf{H}^2$ $(n = 2)$ is horizontal Gaussian mask. $\mathbf{H}^3$ $(n = 3)$ is right diagonal Gaussian mask. $\mathbf{H}^4$ $(n = 4)$ is vertical Gaussian mask and $\mathbf{H}^5$ $(n = 5)$ is left diagonal Gaussian mask. These of all are illustrated in Figures 1 (a) to (e).

The use of the mask is adjusted to the image texture. If the part from the image has a flat texture, it uses a Gaussian mask as in Figure 1(a). While part of the image that has the edges texture in the horizontal direction, vertical, Diagonal Left and Right Diagonal, then successively used Gaussian mask as in Figures 1 (b) to (e). More details in the fifth Gaussian mask design will be explained in the following subsection.

3.2.1. *Design of Gaussian mask.* Gaussian mask can be set using two-dimension Gaussian function as in Equation (7).

$$H = Ae^{-\left(\frac{(x-x_0)^2}{2\sigma_x^2} + \frac{(y-y_0)^2}{2\sigma_y^2}\right)} \tag{7}$$

A is amplitude coefficient. x_0 and y_0 are the coordinates of the center of the mask. σ_x and σ_y are the x and y spreads of the blob. A two-dimensional elliptical Gaussian function can be written in Equation (8).

$$H = Ae^{-\left(a(x-x_0)^2 + 2b(x-x_0)(y-y_0) + c(y-y_0)^2\right)} \tag{8}$$

In this case, A is the amplitude. The shape of the Gaussian blob is determined by variables of a, b and c, and they can be written as Equations (9) to (11), respectively.

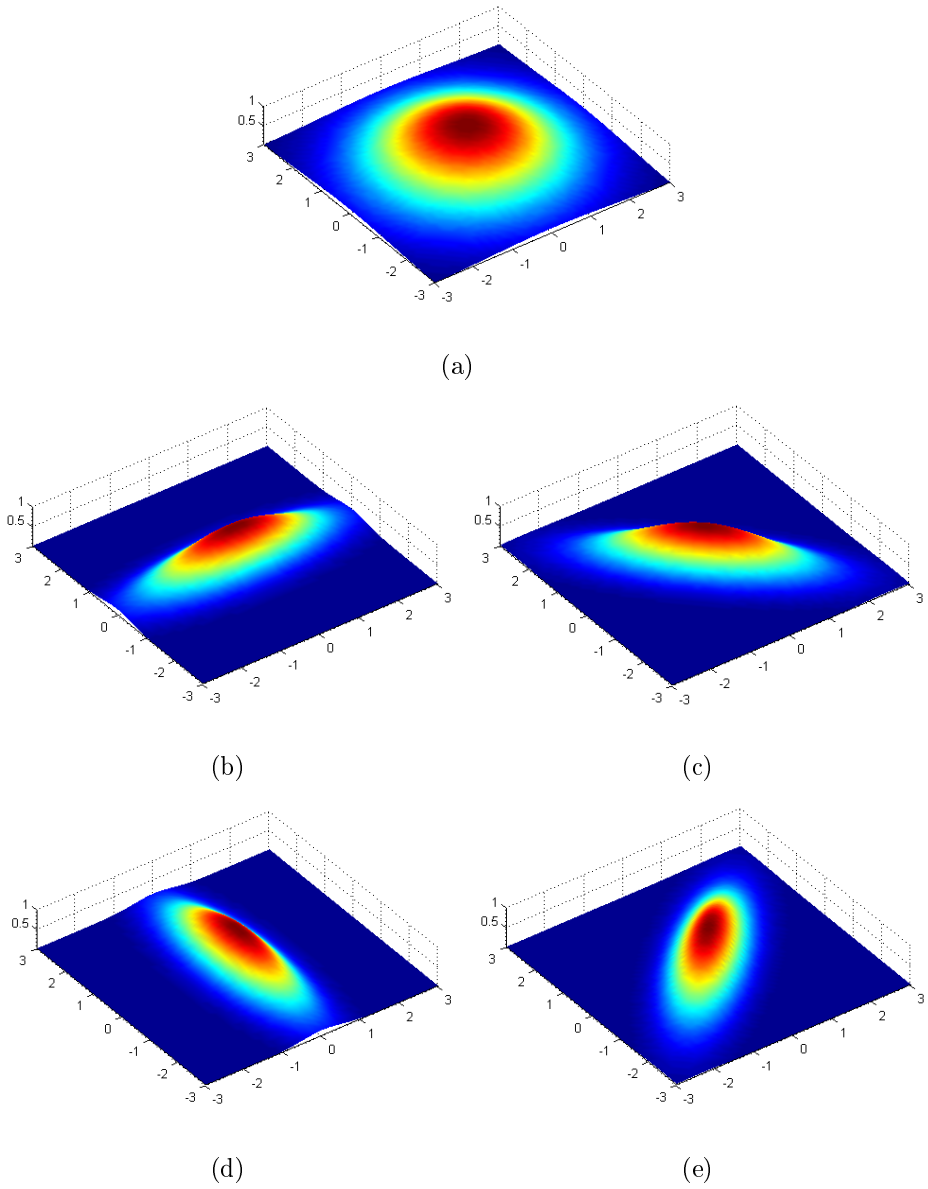
$$a = \frac{\cos^2\theta}{2\sigma_x^2} + \frac{\sin^2\theta}{2\sigma_y^2} \tag{9}$$

$$b = \frac{\sin^2\theta}{4\sigma_x^2} + \frac{\sin^2\theta}{4\sigma_y^2} \tag{10}$$

$$c = \frac{\sin^2\theta}{2\sigma_x^2} + \frac{\cos^2\theta}{2\sigma_y^2} \tag{11}$$

The value of σ_x and σ_y will affect the shape of the Gaussian blob. If both have the same value, there will produce a symmetrical Gaussian curve shape as shown in Figure 1(a).

We set $\sigma_y = 2 \times \sigma_x$ for another mask. There are four θ values (π, $\frac{1}{4}\pi$, $\frac{1}{2}\pi$ and $\frac{3}{4}\pi$), and these will produce four shapes of mask as illustrated in Figure 1(b) to Figure 1(e).



(a)

(b) (c)

(d) (e)

FIGURE 1. Five types of Gaussian mask. (a) Symmetrical Gaussian mask ($\mathbf{H}^1$), (b) Horizontal Gaussian mask ($\mathbf{H}^2$), (c) Right diagonal Gaussian mask ($\mathbf{H}^3$), (d) Vertical Gaussian mask ($\mathbf{H}^4$) and (e) Left diagonal Gaussian mask ($\mathbf{H}^5$).

The total value of the mask element H^n_{xy} is more than one. This will cause the results of filtering tending to contrast. Therefore, we must normalize mask H^n_{xy} in order to have the total elements equal to one. Equation (12) is normalization of mask H_{xy}.

$$\hat{H}^n_{xy} = \frac{H^n_{xy}}{\sum_x \sum_y H^n_{xy}} \tag{12}$$

3.3. **Edge detection.** Edge detection is used to make maps of an edge area in the image. For designing an edge map, we use Sobel kernel. Furthermore, the process is continued by threshold of the edge image. Equations (13) and (14) are the filtering image with horizontal and vertical Sobel kernel. Meanwhile, Equation (15) is the average of

Equations (13) and (14).

$$\mathbf{D}_h = \hat{\mathbf{P}} * \mathbf{H}_h \tag{13}$$

$$\mathbf{D}_v = \hat{\mathbf{P}} * \mathbf{H}_v \tag{14}$$

$$\mathbf{D} = \frac{1}{2}(\mathbf{D}_h + \mathbf{D}_v) \tag{15}$$

$\mathbf{H}_h$ and $\mathbf{H}_v$ are horizontal and vertical Sobel filters, respectively.

By performing threshold process of Equation (15), edge map will be obtained. Equation (16) is the threshold of Equation (15).

$$\hat{\mathbf{D}}_{ij} = \begin{cases} 1 & \text{if } D_{ij} > \tau \\ 0 & \text{others} \end{cases} \tag{16}$$

$\hat{\mathbf{D}}$ is a binary map of an edge image. $\hat{\mathbf{D}}_{ij}$ is an element of an edge image ($\hat{\mathbf{D}}$). D_{ij} is an element of matrix $\mathbf{D}$. In this case, 1 indicates an edge pixel and 0 is not an edge.

3.3.1. *Gaussian mask selection.* In the Gaussian noise removal, we use five Gaussian masks as shown in Figures 1 (a) to (e). Referring to the edge-map of the image, we can select Gaussian mask which is suitable for texture. We present five rules in the selection of Gaussian mask.

1) If not an edge, we use Gaussian mask with probability density function (pdf) as shown in Figure 1(a).
2) If an edge in the horizontal direction, we use Gaussian mask with probability density function (pdf) as shown in Figure 1(b).
3) If an edge in the right-diagonal direction, we use Gaussian mask with probability density function (pdf) as shown in Figure 1(c).
4) If an edge in the vertical direction, we use Gaussian mask with probability density function (pdf) as shown in Figure 1(d).
5) If an edge in the left-diagonal direction, we use Gaussian mask with probability density function (pdf) as shown in Figure 1(e).

The mask can be changed during the convolution process following the edge directions in the image. This convolution is different from conventional filtering process that uses only one mask during the convolution process. Equation (6) is the conditional convolution based on five rules. This is a new convolution process that we call with adaptive convolution based on edge direction.

3.4. **Gaussian filter improvement.** To improve the performance of a Gaussian filter designed, we combine it with the theory of adaptive Wiener [21]. Convolution process of an adaptive window uses a fix mask. Meanwhile, the proposed method uses conditional convolution based on five rules that have been explained in the previous subsection.

Filter Adaptive Wiener with m_F and σ_F^2 has been updated, and it is presented on Equation (17) [21].

$$F(x,y) = m_F(x,y) + \frac{\sigma_F^2(x,y)}{\sigma_F^2(x,y) + \sigma_v^2)} \times \left(\hat{P}(x,y) - m_F(x,y)\right) \tag{17}$$

σ_F^2 is the power spectrum of the element matrix from representation of the original image. σ_v^2 is the power spectrum of noise prediction that is obtained from averaging of σ_F^2. m_F is mean of original image. In this case, $m_F \simeq \hat{F}$. Furthermore, σ_F^2 can be obtained using Equation (18).

$$\sigma_F^2 = \left(\hat{P}^2 * H^n\right) - m_F^2 \tag{18}$$

In Equation (17), the local mean remains unmodified. Meanwhile, the local contrast is scaled according to the relative amplitudes of σ_F^2 and σ_v^2. If σ_F^2 is higher than σ_v^2, local contrast of $\hat{P}(x, y)$ was assumed in accordance with the $F(x, y)$ and local contrast of $\hat{P}(x, y)$ is not reduced, and vice versa.

4. **Experimental Result and Discussion.** We have conducted some experiments to see the performance of the proposed method in the qualitative and quantitative parameters. Different densities of Gaussian plus impulse noise have been tested in our research. We symbolize impulse noise density by p and Gaussian noise density by (σ). In our experiment, we use high impulse noise densities ($p = 50\%$, 70% and 90%). Furthermore, we use Gaussian noise density ($\sigma = 5$ and 10). The entire method was implemented in CPU 3.3 GHz and RAM 4GB using MATLAB 7.5.0 release 2007b.

4.1. **Experimental result.** We compare our proposed method with the previous studies. The previous methods to compare with our proposed method are Two-phase method (TP) [7], Fast two-phase image deblurring (FTPID) [8], Total variation (TV) [11] and Modifying K-SVD [20]. In order to evaluate performance of the filter, a simulation result was tested by using Lena's image (512×512) and Barbara's image (512×512). Figure 2 is the filtering result of Lena's image with $\sigma = 10$ and $p = 70\%$, and Figure 3 is the filtering result of Barbara's image with $\sigma = 5$ and $p = 90\%$.

4.2. **Evaluation method.** The quality of the filtering image is evaluated by qualitative and quantitative evaluation. The information about these evaluations is explained in subsection below.

FIGURE 2. The filtering result of the Lena corrupted image. (a) Lena image is corrupted by Gaussian and impulse noise with $\sigma = 10$ and $p = 70\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD [20]. (f) Proposed method.

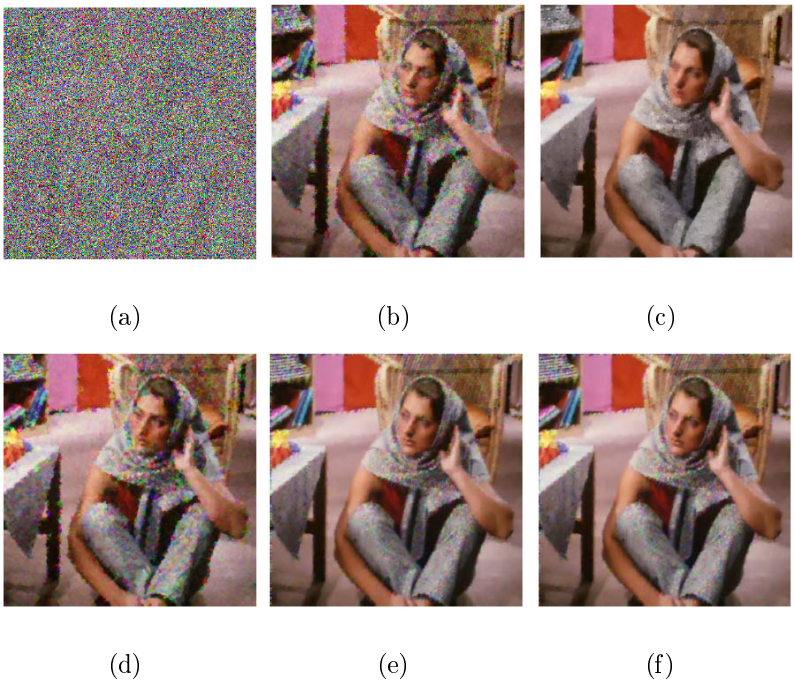


FIGURE 3. The filtering result of the Barbara corrupted image. (a) Barbara corrupted by Gaussian and impulse noise with $\sigma = 5$ and $p = 90\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD [20]. (f) Proposed method.

4.2.1. *Qualitative evaluation.* The quality of filtering image result is evaluated by visual observation.

The corrupted image by $\sigma = 10$ and $p = 70\%$ is shown in Figure 2(a). In this condition, it has difficulty in identifying an object in the image. Further, Figures 2(b), 2(c) and 2(d) are the filtering results of TP, FTPID and TV methods, respectively. The quality filters of TP, FTPID and TV methods are not good enough to reduce the mixed Gaussian plus impulse noise in condition that $\sigma = 10$ and $p = 70\%$. The result of modifying K-SVD is presented in Figure 2(e). By visual observation, the capability to reduce impulse noise in Figure 2(e) is a little bit better than Figures 2(b), 2(c) and 2(d). However, the quality of the filtering result by modifying K-SVD as shown in Figure 2(e) is too smooth. The filtering result of the proposed method is presented in Figure 2(f). The proposed method looks better than other comparison methods. The detail of important pixel has maintained in the proposed method.

The corrupted image by $\sigma = 5$ and $p = 90\%$ is shown in Figure 3(a). Further, Figures 3(b), 3(c) and 3(d) are the filtering results of TP, FTPID and TV methods, respectively. Similar with Lena's image, the quality filters of TP, FTPID and TV methods are not good enough to reduce the mixed Gaussian plus impulse noise in condition that $\sigma = 5$ and $p = 90\%$. The result of modifying K-SVD is presented in Figure 3(e). By visual observation, the capability to reduce impulse noise in Figure 3(e) is a little bit better than Figures 3(b), 3(c) and 3(d). Meanwhile, Figure 3(f) is the filtering result of the proposed method. Figure 3(f) looks better than Figure 3(e). In this case, the lines door behind the chair in Figure 3(f) is smoother than Figure 3(e).

In addition, we use SSIM-map in the qualitative evaluation. SSIM-map is a local perceptual quality indicator that is used to measure the similarity between original image

and the filtering result. If SSIM-map has a high-contrast result, the filtering image results also have a high similarity with the original image. Meanwhile, SSIM-map has a low-contrast result when the filtering image results have a little similarity with the original image.

Figure 4 is SSIM-map result of Figure 2. Dominant black color is resulted from the corrupted Lena's image as shown in Figure 4(a). Figures 4(b) and 4(c) are the SSIM-map of TP and FTPID. SSIM-map of both methods is causing low-contrast quality. Further, SSIM-map of TV method is shown in Figure 4(d). By visual observation, Figure 4(d) has a higher contrast image than Figures 4(a), 4(b), and 4(c). Meanwhile, Figures 4(e) and 4(f) are SSIM-map from modifying K-SVD method and the proposed method. Figures 4(e) and 4(f) have the quality a little bit similar. However, Figure 4(f) has a higher contrast quality than Figure 4(e).

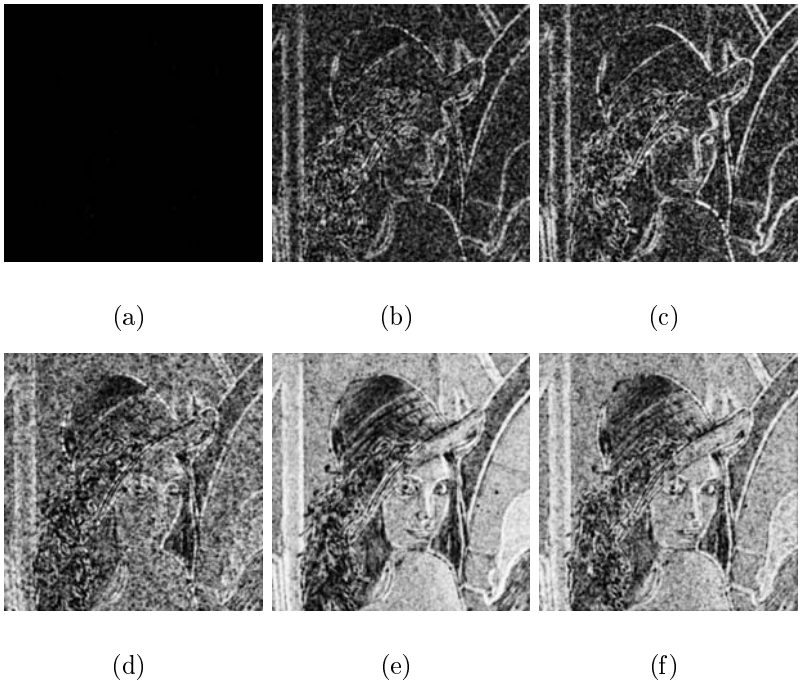


FIGURE 4. SSIM-map of Lena image in Figure 2. (a) Lena image corrupted by Gaussian and impulse noise with $\sigma = 10$ and $p = 70\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD method [20]. (f) Proposed method.

Figure 5 is SSIM-map result of Figure 3 Barbara's image. Dominant black color is resulted from the corrupted Barbara's image as shown in Figure 5(a). Figures 5(b), 5(c) and 5(d) are the SSIM-map of TP, FTPID and TV methods. These methods have low-contrast quality in the SSIM-map result. Meanwhile, Figures 5(e) and 5(f) are SSIM-map from modifying K-SVD and the proposed method. Both methods have the quality a little bit similar. However, the proposed method in Figure 5(f) has a higher contrast image than Figure 5(e), especially in the face area of Barbara's image.

4.2.2. *Quantitative evaluation.* The important evaluations to determine the quality of image filtering results are PSNR, SSIM index and computation time result. PSNR is an abbreviation of Peak Signal-to-Noise Ratio that uses a standard mathematical model. The PSNR is the ratio of the maximal power of original image and the noise power of

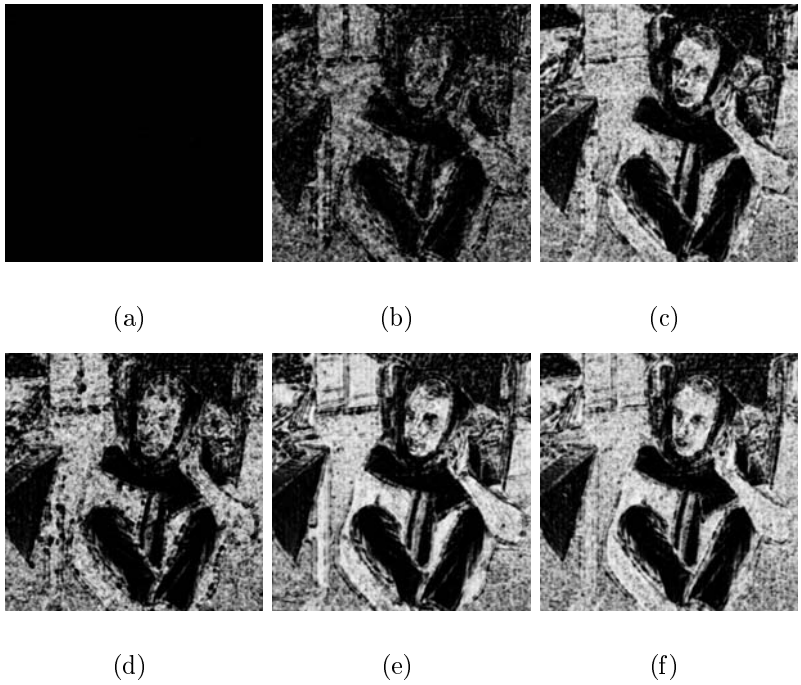


FIGURE 5. SSIM-map of Barbara image in Figure 3. (a) Barbara image corrupted by Gaussian and impulse noise with $\sigma = 5$ and $p = 90\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD [20]. (f) Proposed method.

distorted image. The bigger the PSNR value is, the better image quality is and vice versa. Peak signal-to-noise ratio (PSNR) is illustrated in Equation (19).

$$\text{PSNR} = 20 \log \frac{255.M.N}{\sum_{j=1}^{M} \sum_{i=1}^{N} (x_{ij} - y_{ij})^2} \tag{19}$$

PSNR is usually expressed in terms of a logarithmic decibel scale as illustrated in Equation (19). M is represented as row of an image; N is a column of an image; x_{ij} is an original image and y_{ij} is the filtering result.

The PSNR values of the several methods with different various impulse noise densities ($p = 50\%$, 70% and 90%) and standard deviation ($\sigma = 5$ and 10) are presented in Table 1.

Referring to Table 1, PSNR values for Barbara's and Lena's images between modifying K-SVD method and our proposed method are almost similar. In this case, our proposed method is a little higher than in the modifying K-SVD method. Sometimes, the proposed method also has PSNR value little bit lower than the modifying K-SVD. However, if we analyze to use computation time parameter, the proposed method is faster than the modifying K-SVD method.

Further, the structural similarity (SSIM) index is a method for measuring the similarity between two images. MSSIM takes the idea that human vision is sensitive to structural distortion of nature images, so local structure distortion as well as luminance and contrast are considered in MSSIM index [22]. The MSSIM converges to 1 [23]. It will obtain a better image quality. The MSSIM index values exhibit much better analysis with the

TABLE 1. Comparison result of PSNR value

Images	Noise p and σ	Methods				
		Cai1 [7]	Cai2 [8]	TV [11]	Modifying K-SVD [20]	Proposed
Barbara	50 and 5	25.47	27.86	25.77	28.73	28.76
Peppers		27.47	29.39	28.21	30.24	30.87
House		29.29	31.82	31.52	34.33	34.33
Lena		28.81	31.25	30.11	32.40	32.47
Barbara	70 and 5	22.91	25.89	23.95	26.09	26.25
Peppers		24.74	27.75	26.03	28.02	28.54
House		26.21	30.06	28.03	30.47	31.23
Lena		26.08	29.46	27.54	29.72	30.28
Barbara	90 and 5	19.53	23.28	20.61	22.45	23.38
Peppers		20.86	23.52	21.96	23.63	25.54
House		21.09	25.54	22.75	23.79	26.73
Lena		21.90	26.29	23.28	25.43	26.97
Barbara	50 and 10	24.21	26.35	25.30	28.10	28.10
Peppers		25.60	27.42	27.44	29.41	29.89
House		26.56	28.74	29.94	33.17	33.17
Lena		26.37	28.38	28.92	31.07	31.77
Barbara	70 and 10	22.08	25.08	23.51	25.73	25.75
Peppers		23.57	26.65	25.32	27.55	27.86
House		24.38	28.32	26.96	29.82	30.05
Lena		24.51	27.83	26.59	29.17	29.78
Barbara	90 and 10	19.66	22.76	20.29	22.50	23.33
Peppers		20.31	23.04	21.52	23.74	24.71
House		20.41	24.87	22.24	23.99	25.17
Lena		21.12	25.42	22.70	25.27	26.33

qualitative visual appearance that is calculated by using Equation (20).

$$\mathrm{MSSIM}(f, f_o) = \frac{1}{M}\sum_{j=1}^{M}\mathrm{SSIM}(f_j, f_{o_j}) \tag{20}$$

f and f_o are the clean image and filtered images, respectively. f_j and f_{o_j} are the image contents at the local window. Furthermore, M is the number of local windows in the image [24]. Furthermore, Equation (21) is used to obtain SSIM.

$$\mathrm{SSIM}(f, f_o) = \frac{(2\mu_f\mu_{f_o} + C_1)(2\sigma_{ff_o} + C_2)}{(\mu_f^2 + \mu_{f_o}^2 + C_1)(\sigma_f^2 + \sigma_{f_o}^2 + C_2)} \tag{21}$$

μ_f and μ_{f_o} are the mean intensities of images f and f_o respectively. σ_f and σ_{f_o} are standard deviations of images f and f_o respectively. C_1 and C_2 are constant to avoid instability.

Table 2 presents MSSIM index results. Similar to PSNR results, our proposed method is a little higher than the modifying K-SVD method. However, in general, the quality of our proposed method is better than the modifying K-SVD and all comparison methods (TP, FTPID and TV methods).

Furthermore, referring to the computation time result our proposed method as presented in Table 3 has faster results compared to all methods (TP, FTPID, TV and Modifying K-SVD). As you know, the fast computing time and a good visual result are very important task in the image filtering process.

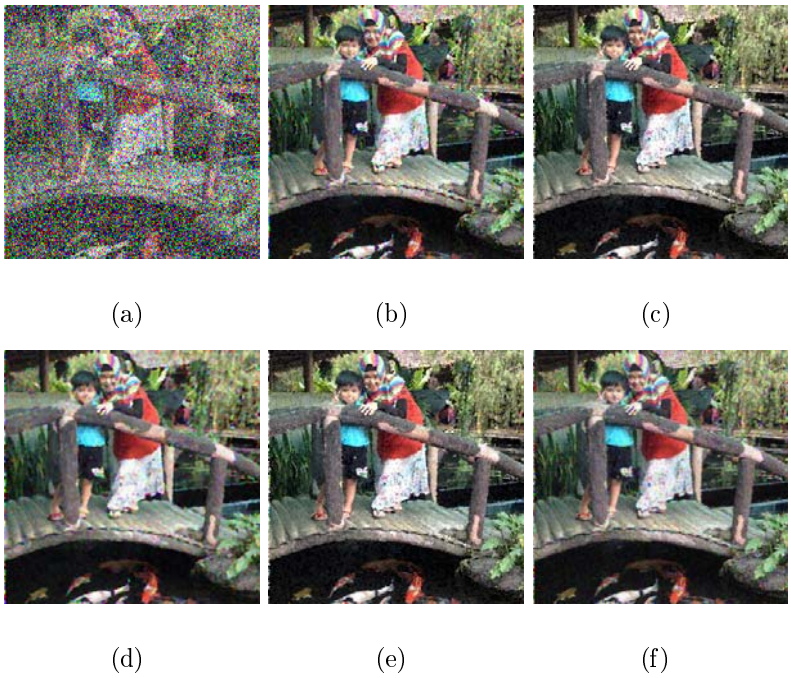
TABLE 2. Comparison result of MSSIM index value

Images	Noise p and σ	Methods				
		Cai1 [7]	Cai2 [8]	TV [11]	Modifying K-SVD [20]	Proposed
Barbara	50 and 5	0.85	0.83	0.80	0.90	0.90
Peppers		0.74	0.80	0.77	0.80	0.82
House		0.79	0.85	0.89	0.93	0.93
Lena		0.86	0.85	0.88	0.91	0.92
Barbara	70 and 5	0.76	0.78	0.75	0.83	0.84
Peppers		0.65	0.77	0.71	0.76	0.78
House		0.72	0.85	0.84	0.90	0.93
Lena		0.79	0.83	0.84	0.88	0.90
Barbara	90 and 5	0.56	0.67	0.62	0.69	0.71
Peppers		0.52	0.69	0.60	0.68	0.73
House		0.58	0.81	0.71	0.79	0.84
Lena		0.63	0.77	0.73	0.79	0.83
Barbara	50 and 10	0.75	0.7	0.77	0.86	0.86
Peppers		0.57	0.65	0.70	0.75	0.76
House		0.57	0.66	0.81	0.85	0.85
Lena		0.73	0.67	0.84	0.87	0.88
Barbara	70 and 10	0.66	0.68	0.71	0.78	0.78
Peppers		0.50	0.66	0.64	0.73	0.74
House		0.53	0.70	0.74	0.83	0.84
Lena		0.67	0.7	0.79	0.85	0.87
Barbara	90 and 10	0.55	0.62	0.59	0.66	0.69
Peppers		0.42	0.64	0.53	0.66	0.68
House		0.42	0.74	0.62	0.78	0.80
Lena		0.54	0.71	0.68	0.76	0.79

4.3. **Discussion.** Proposed method has a filtering result slightly better than the complex methods such as K-SVD. The proposed method is 118.75 times faster than K-SVD for denoising a Lena image with impulse noise density 50 and Gaussian noise variance 5. So, the proposed method is more possible implemented for denoising corrupted image sequences by mixed Gaussian and impulse noise. This caused that the proposed method is a much simpler method and faster in the processing time than the previous method such as illustrated in Table 3. On the application of image sequences, we first take a frame as an image, then we apply a filter to that image, and at last we convert back into a movie. For implementing on real time image sequences denoising, the faster hardware is needed.

The method is made not only good implemented on the image in an indoor area but also for the image of the outdoors. We give examples for the images in an outdoor area as Figure 6 and Figure 7. The simulation results in Figures 6 and 7 show that the simple method of proposed method is optimal for reducing noise compared to the comparison methods. The evaluation of Figure 6 obtains PSNR for Cai1 [7] = 18.92 dB, Cai2 [8] = 20.54 dB, TV [11] = 19.31 dB, Modifying K-SVD [20] = 20.22 dB and Proposed method = 20.90 dB.

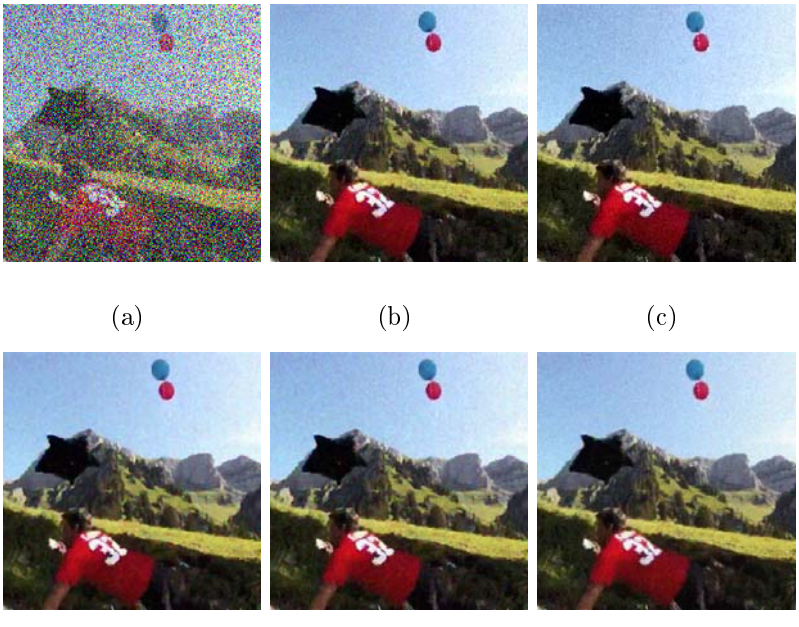
5. **Conclusions.** Two stages filtering process is proposed for reducing the mixed Gaussian plus impulse noise. We make impulse noise removal of a merger between the decision based method and the kernel observation. Furthermore, reducing Gaussian noise in the



(a) (b) (c)

(d) (e) (f)

FIGURE 6. The filtering result for outdoor image. (a) Image is corrupted by Gaussian and impulse noise with $\sigma = 10$ and $p = 50\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD [20]. (f) Proposed method.



(a) (b) (c)

(d) (e) (f)

FIGURE 7. The filtering result for sequence images. (a) Image is corrupted by Gaussian and impulse noise with $\sigma = 10$ and $p = 50\%$. (b) TP method [7]. (c) FTPID method [8]. (d) TV method [11]. (e) Modifying K-SVD [20]. (f) Proposed method.

TABLE 3. Comparison result of times process in second

Images	Noise p and σ	Methods				
		Cai1 [7]	Cai2 [8]	TV [11]	Modifying K-SVD [20]	Proposed
Barbara	50 and 5	9.63	22.09	19.78	1271.93	5.43
Peppers		9.69	20.84	21.75	540.44	5.20
House		4.17	4.32	3.06	163.87	1.33
Lena		9.59	17.85	15.43	621.10	5.23
Barbara	70 and 5	10.67	33.37	23.35	996.78	6.08
Peppers		10.51	30.48	28.97	400.52	5.73
House		3.88	5.91	3.45	149.76	1.42
Lena		11.17	26.54	20.26	468.57	5.84
Barbara	90 and 5	40.64	86.66	63.74	573.56	7.89
Peppers		42.14	84.41	71.79	302.44	8.05
House		11.36	16.80	10.95	157.28	1.85
Lena		40.00	73.10	58.24	343.16	7.55
Barbara	50 and 10	9.34	25.08	19.52	392.50	5.44
Peppers		9.63	22.62	21.36	181.17	5.30
House		3.73	4.40	2.50	92.45	1.22
Lena		9.59	19.20	15.30	230.53	5.33
Barbara	70 and 10	10.20	33.74	21.06	294.89	6.05
Peppers		11.43	34.79	25.93	195.88	6.10
House		3.92	6.86	3.56	89.26	1.40
Lena		11.34	28.80	19.06	269.60	6.04
Barbara	90 and 10	46.25	85.54	59.94	157.85	7.60
Peppers		42.42	82.15	69.14	133.91	7.93
House		10.94	17.43	10.59	83.33	1.81
Lena		40.00	74.71	58.22	127.95	7.63

image can be conducted by convolution of the corrupted image with a Gaussian mask. The proposed method provides five Gaussian masks in different directions (symmetrical, horizontal, vertical, diagonal left-diagonal and right-diagonal Gaussian). Simulation result is able to achieve significant performance improvements of the previous study as the comparison result.

For future work, we will develop the capability of the proposed method not only optimal to reduce the mixed Gaussian plus impulse noise, but also for other types of noise.

REFERENCES

[1] F. Utaminingrum, K. Uchimura and G. Koutaki, Optimization Gaussian noise removal using hybrid filter based on mean impulse fuzzy and fuzzy aliasing filter methods, *IEEJ Trans. Electronics, Informations and System*, vol.133, no.1, pp.150-158, 2012.

[2] A. Chaudhry, J. Y. Kim and T. A. Tuan, Improved adaptive fuzzy punctual kriging filter for image restoration, *International Journal of Innovative Computing, Information and Control*, vol.9, no.2, pp.583-598, 2013.

[3] F. Utaminingrum, K. Uchimura and G. Koutaki, Speedy filters for removing impulse noise based on an adaptive window observation, *International Journal of Electronics and Communications (AEU)*, vol.69, pp.95-100, 2015.

[4] S. Masood, A. Hussain, M. A. Jaffar, A. M. Mirza and T. Choi, Fuzzy random impulse noise reduction technique based on statistical estimators, *International Journal of Innovative Computing, Information and Control*, vol.7, no.6, pp.3395-3406, 2011.

[5] E. J. Leavline, S. Sutha and D. A. A. G. Singh, On the suitability of multiscale image representation schemes as applied to noise removal, *International Journal of Innovative Computing, Information and Control*, vol.10, no.3, pp.1135-1147, 2014.

[6] K. K. V. Toh and N. A. M Isa, Augmented variational series and histogram-based clustering for universal impulse denoising, *International Journal of Innovative Computing, Information and Control*, vol.8, no.9, pp.5969-5998, 2012.

[7] J.-F. Cai, R. Chan and M. Nikolova, Two-phase methods for deblurring images corrupted by impulse plus gaussian noise, *Inverse Problem Imaging*, vol.2, no.2, pp.187-204, 2008.

[8] J.-F. Cai, R. Chan and M. Nikolova, Fast two-phase image deblurring under impulse noise, *Journal of Mathematical Imaging and Vision*, vol.36, pp.46-53, 2010.

[9] B. Li, Q. Liu, J. Xu and X. Luo, A new method for removing mixed noises, *Sci. China Inf. Sciences*, vol.54, pp.51-59, 2011.

[10] Y. Xiao, T. Zeng, J. Yu and M. Ng, Restoration of images corrupted by mixed Gaussian-impulse noise via l1-l0 minimization, *Pattern Recog.*, vol.44, pp.1708-1720, 2011.

[11] P. Rodriguez, R. Rojas and B. Wohlberg, Mixed Gaussian impulse noise image restoration via total variation, *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp.1077-1080, 2012.

[12] D. Chen, H. Zhang and L. Cheng, Nonlocal variational model and filter algorithm to remove multiplicative noise, *Opt. Eng.*, vol.49, no.7, 2010.

[13] G. Aubert and J. F. Aujol, A variational approach to remove multiplicative noise, *SIAM J. Appl. Math.*, vol.68, no.4, pp.925-946, 2008.

[14] J. Shi and S. Osher, A nonlinear inverse scale space method for a convex multiplicative noise model, *SIAM J. Appl. Math.*, vol.1, no.3, pp.294-321, 2008.

[15] G. Steidl and T. Teuber, Removing multiplicative noise by Douglas-Rachford splitting methods, *J. Math. Imaging Vis.*, vol.36, no.2, pp.168-184, 2010.

[16] J. Zhang, Z. Wei and L. Xiao, A fast adaptive reweighted residual-feedback iterative algorithm for fractional-order total variation regularized multiplicative noise removal of partly-textured images, *Signal Processing*, vol.98, pp.381-395, 2014.

[17] A. Buades, B. Coll and J. Morel, A review of image denoising algorithms with a new one, *Multi-scale Modelling Simulation*, vol.4, no.2, pp.490-530, 2005.

[18] M. Aharon, M. Elad and A. Bruckstein, K-SVD: An algorithm for denoising overcomplate dictionaries for sparse representation, *IEEE Trans. Image Processing*, pp.9-12, 2005.

[19] M. Elad and M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, *IEEE Trans. Image Processing*, vol.15, no.12, pp.3736-3745, 2006.

[20] F. Utaminingrum, K. Uchimura and G. Koutaki, Modifying K-SVD for optimization of the mixed gaussian and impulse noise removal, *International Conference of the 20th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, Okinawa, Japan, pp.191-197, 2014.

[21] L. S. Jae, *Two-dimensional Signal and Image Processing*, 1st Edition, PTR Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1990.

[22] F. Utaminingrum, K. Uchimura and G. Koutaki, Dual filter based on pipeline models for removing high density impulse noise, *International Workshop on Advanced Image Technology (IWAIT)*, Bangkok, Thailand, pp.522-527, 2014.

[23] Q. Ma and L. Zhang, Image quality assessment with visual attention, *IEEE Pattern Recognition*, pp.1-4, 2008.

[24] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Trans. Image Processing*, vol.13, no.4, pp.600-612, 2004.